

DBIF Tag & Channel Mechanism

Integrated Test & Operations System

Developer Documentation

5 October 2006

Copyright 1999-2006, United States Government as represented by the Administrator of the National Aeronautics and Space Administration. No copyright is claimed in the United States under Title 17, U.S. Code.

This software and documentation are controlled exports and may only be released to U.S. Citizens and appropriate Permanent Residents in the United States. If you have any questions with respect to this constraint contact the GSFC center export administrator, <Thomas.R.Weisz@nasa.gov>.

This product contains software from the Integrated Test and Operations System (ITOS), a satellite ground data system developed at the Goddard Space Flight Center in Greenbelt MD. See <<http://itos.gsfc.nasa.gov/>> or e-mail <itos@itos.gsfc.nasa.gov> for additional information.

You may use this software for any purpose provided you agree to the following terms and conditions:

1. Redistributions of source code must retain the above copyright notice and this list of conditions.
2. Redistributions in binary form must reproduce the above copyright notice and this list of conditions in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
This product contains software from the Integrated Test and Operations System (ITOS), a satellite ground data system developed at the Goddard Space Flight Center in Greenbelt MD.

This software is provided 'as is' without any warranty of any kind, either express, implied, or statutory, including, but not limited to, any warranty that the software will conform to specification, any implied warranties of merchantability, fitness for a particular purpose, and freedom from infringement and any warranty that the documentation will conform to their program or will be error free.

In no event shall NASA be liable for any damages, including, but not limited to, direct, indirect, special or consequential damages, arising out of, resulting from, or in any way connected with this software, whether or not based upon warranty, contract, tort, or otherwise, whether or not injury was sustained by persons or property or otherwise, and whether or not loss was sustained from or arose out of the results of, or use of, their software or services provided hereunder.

Log Widget

The Log widget is a widget for managing a message queue. The Log widget is particularly good at managing event messages.

Synopsis

```
#include <Log.h>

widget = XtVaCreateManagedWidget("widget-name",
                                   logWidgetClass, parent,
                                   va-arglist);
```

Description

The Log widget manages a message queue. Each message is a line in the view and is a *String* which may contain escape sequences to control the colors used to display the message or to sound the bell. '\n' characters within messages are converted to spaces (messages need not end in '\n') and tabs are expanded assuming 8-character tab stops.

When the Log widget is created, the message queue is empty. Messages are added to the message queue via the programmatic interface function *LogAppend()*. When a message is added it goes as the bottom line in the queue. If the queue is full, the oldest message (the top line) is deleted before the new message is added.

Normally the entire message queue cannot be displayed in the Log widget's window. The Log window provides vertical and horizontal scrollbars so the entire message queue can be viewed.

The Log widget operates in either of two modes: In autoscroll mode the view automatically scrolls so that, as new messages arrive, the newest message is visible. In scrollbar mode, the view does not scroll as new messages get appended unless the topmost messages got purged.

The Log widget is a composite widget with three children: *XmScrollBar hbar*, *XmScrollBar vbar*, and *XmDrawingArea view*. The Log widget uses the motif library.

Resources

NAME	CLASS	TYPE	DEFAULT VALUE	ACCESS
autoscroll	Autoscroll	Boolean	TRUE	CSG
bellrate	Bellrate	int	1000	CSG
font	Font	XFontStruct *	XtDefaultFont	C G
fontColor0	FontColor	Pixel	XtDefaultBackground	C G
fontColor1	FontColor	Pixel	red	C G
fontColor2	FontColor	Pixel	dark green	C G
fontColor3	FontColor	Pixel	yellow	C G
fontColor4	FontColor	Pixel	blue	C G
fontColor5	FontColor	Pixel	magenta	C G

fontColor6	FontColor	Pixel	cyan	C G
fontColor7	FontColor	Pixel	XtDefaultForeground	C G
hbar	Scrollbar	Widget	dynamic	G
hideHoriz	HideScrollbar	Boolean	FALSE	CSG
leftMargin	Margin	int	4	C G
queueWidth	QueueWidth	int	80	C G
queueHeight	QueueHeight	int	24	C G
topMargin	Margin	int	4	C G
vbar	Scrollbar	Widget	dynamic	G
view	DrawingArea	Widget	dynamic	G
viewWidth	ViewWidth	int	1	C G
viewHeight	ViewHeight	int	1	C G

(Access: C – resource may be set at creation time; G – resource may be retrieved using `XtVaGetValues`; and S – resource may be set using `XtVaSetValues`).

autoscroll Controls autoscroll mode vs. scrollbar mode.

When *autoscroll* is TRUE, the Log widget is in autoscroll mode. The vertical scrollbar is insensitive and scrolling is automatic, so that, as new messages arrive, the newest message is visible. Setting *autoscroll* TRUE scrolls so the bottommost message is visible.

When *autoscroll* is FALSE, the Log widget is in scrollbar mode. The vertical scrollbar is sensitive and controls scrolling. There is no automatic scrolling unless the topmost messages got purged.

bellrate Controls how frequently the alarm may sound. If the alarm has not sounded in the previous *bellrate* milliseconds and a '\a' character is encountered in the newly arrived messages, the alarm will sound.

font Must be a fixed width font.

fontColor0 ... *fontColor7*

The colors to use when displaying messages. Initially messages are drawn using *fontColor7* as the foreground color and *fontColor0* as the background. This is overridden by messages containing these escape sequences:

Esc [0 m Reset. Sets foreground to *fontColor7* and background to *fontColor0*.

Esc [3 0 m ... Esc [3 7 m
Sets the foreground color.

Esc [4 0 m ... Esc [4 7 m
Sets the background color.

hbar The widget ID of the horizontal scrollbar widget (class *XmScrollBar*).

hideHoriz Controls whether or not the horizontal scrollbar is visible.

leftMargin

queueWidth

queueHeight

The size, in characters and lines, of the message queue. These resources can't be changed via `XtVaSetValues()`.

